REMARKS/ARGUMENTS

Prior to the entry of this Amendment, claims 1-24 were pending in this application. Claim 1 is amended, no claims are added, and no claims are canceled herein. Therefore, claims 1-24 remain pending. Applicants respectfully request entry of the amendments and reconsideration of these claims for at least the reasons presented below.

Replacement Specification

As requested by the Examiner, a Replacement Specification was re-submitted with the previous Amendment filed April 14, 2006. No indication has yet been given whether this specification has been entered or is acceptable. The Applicants respectfully request entry of this specification if it has not yet been entered.

Title

In the previous Amendments filed April 14, 2006 and December 5, 2005, the Applicants requested the title to be changed to "Responding To Requests For Remote Operation In Systems With Multiple Servers." No indication has yet been given whether this title is acceptable. The Applicants respectfully request that the title be changed as requested or, if this title is not acceptable, the Applicants be so informed and given a chance to submit a revised title.

35 U.S.C. § 102 Rejection, Freeman

Claims 1-24 have been rejected under 35 U.S.C. § 102(e) as being anticipated by U. S. Patent No. 6,785,726 to Freeman et al. (hereinafter "Freeman"). The Applicants respectfully submit the following arguments pointing out significant differences between claims 1-24 submitted by the Applicants and Freeman.

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." MPEP 2131 citing *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053

(Fed. Cir. 1987). Applicants respectfully argue that Freeman fails to disclose each and every claimed element. For example, Freeman fails to disclose, either expressly or inherently, causing the destination system to block new requests. Rather, Freeman teaches either starting a new thread to concurrently handle events in a multi-threaded system or queuing events to be handled, presumably, in the order received in a single-threaded system. In either case, Freeman teaches that new requests will be received by the destination system while previous requests are being processed.

As noted previously, Freeman "relates to a method and apparatus for delivering events to local and remote servers." (Column 1, lines 45-46.) Freeman teaches "a first server having a first plurality of subsystems and a first event bus and a second server having a second plurality of subsystems and a second event bus." (Column 1, lines 49-51.) The method of Freeman includes "posting to the first event bus, by one of the first plurality of subsystems, a local event having as its target another of the first plurality of subsystems [and] . . . posting to the first event bus, by one of the first plurality of subsystems, a remote event having as its target one of the second plurality of subsystems." (Column 1, lines 52-57.) These events include request events "that send a request for service or functionality to another subsystem on the same server or to a remote server in the server farm" and reply events that "occur in response to request events." (Column 27, lines 56-65.) Freeman also discloses a "SendEventandWait" command for issuing a request event. (Column 30, line 60, through column 32, line 36.)

Under Freeman, a source subsystem can issue a request using the "SendEventAndWait" command and then "block further execution of the thread that generated the request event until the response from the destination subsystem is received." (Column 30, line 65, through column 31, line 7.) After the request is issued, a target can be located from a service locator (column 31, lines 10-24) and the request can be sent to the target/destination system via the transport layer and event bus (column 31, lines 25-43). The way in which the destination system handles the event depends on whether the destination system is single-threaded or multi-threaded. (Column 31, lines 41-43.) If the destination subsystem is multi-

threaded, the destination system executes an appropriate handler routine for responding to the request event. (Column 31, lines 44-50.) If the destination subsystem is single-threaded, the destination system places a pointer to the event buffer holding the request event in the event queue associated with the destination subsystem. (Column 31, lines 51-59.)

That is, upon receiving an event, the destination system of Freeman either handles the event immediately or places the event in a queue for later handling depending on whether the system is multi-threaded or single-threaded. However, Freeman does not disclose causing the destination system to block new requests. To the contrary, Freeman teaches either starting a new thread to concurrently handle events in a multi-threaded system or queuing events to be handled, presumably, in the order received in a single-threaded system. In either case, Freeman teaches that new requests will be received while previous requests are being processed. Therefore, Freeman does not teach the destination system blocking new requests.

Claim 1, upon which claims 2-10 depend, recite in part the "remote server blocking all new requests in response to said requests." Freeman does not disclose the remote server, i.e., destination system, blocking all new requests in response to a request. Rather, Freeman teaches either starting a new thread to concurrently handle events in a multi-threaded system or queuing events to be handled, presumably, in the order received in a single-threaded system. In either case, Freeman teaches that new requests will be received by the destination system while previous requests are being processed. For at least these reasons, claims 1-10are distinguishable from Freeman and should be allowed.

Claim 11, upon which claims 12-17 depend, and claim 18, upon which claims 19-24 depend, each recite in part the "remote server blocking new requests in response to said requests." Freeman does not disclose the remote server, i.e., destination system, blocking new requests in response to a request. Rather, Freeman teaches either starting a new thread to concurrently handle events in a multi-threaded system or queuing events to be handled, presumably, in the order received in a single-threaded system. In either case, Freeman teaches

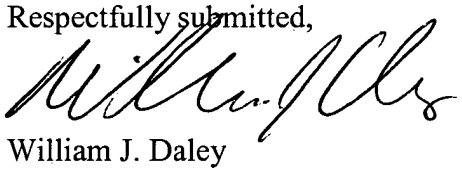
that new requests will be received by the destination system while previous requests are being processed. For at least these reasons, claims 11-24 are distinguishable from Freeman and should be allowed.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 303-571-4000.

Respectfully submitted,



Date: April 14, 2006

William J. Daley
Reg. No. 52,471

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, CA 94111-3834
Tel: 303-571-4000 (Denver)
Fax: 303-571-4000 (Denver)

WJD/sbm

60765216 v1